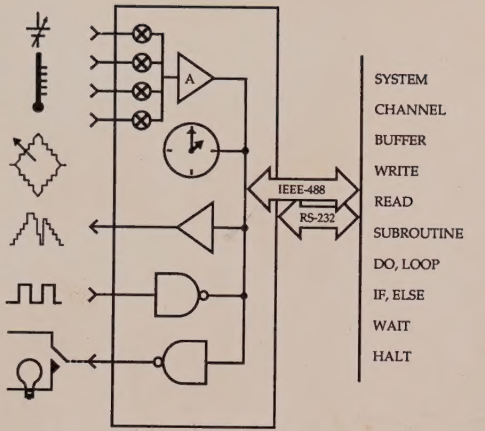
KEITHLEY DAC

Model 576 High Speed Data Logging System

Programming Reference Guide

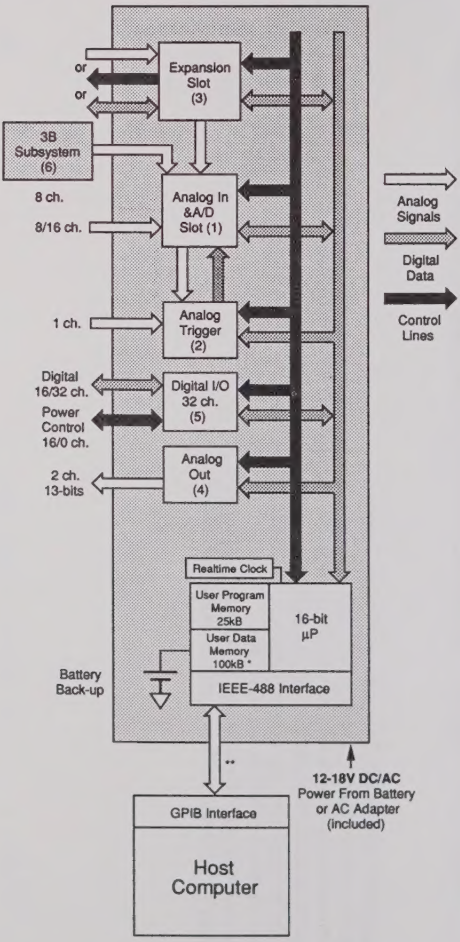


ANDY LARAI - 1-508-880-3000

All Keithley product names are trademarks or registered trademarks of Keithley Instruments, Inc.
Other brand and product names are trademarks or registered trademarks of their respective holders.

SECTION 1 - Introduction

- Higher speed, more data storage than conventional data loggers.
- English-like, firmware-resident commands support conditional testing, subroutines, and data management.
- Compact, powerful programs can be downloaded, stored, and executed independent of the computer.
- On-board real-time clock/calendar for time/date stamping and time triggering.
- Expansion slot for transducer signal conditioning or additional I/O channels.
- Operates from 12-18V AC/DC, or 115/230V AC.
- GPIB interface for compatibility with widest range of computers.
- Serial accessory for operation with computers having an RS-232 port.



* 480k factory-installed optional
** or use serial adapter to computer's RS-232 port

Digital reading - 1 byte
Analog reading - 2 bytes
Thermo reading - 4 bytes

ANALOG INPUT

Standard: 16 single-ended or 8 differential channels, plus 8 s.e. inputs compatible with 3B signal conditioning sub-system. Analog trigger circuit operates with analog input for high-speed, oscilloscope-like triggering.

Module	Channels	F.S. Ranges*	Notes
AMM1	16/8	10/1/0.1V**	576-1 – 12-bit, 62.5kHz A/D (3-1/2 digit)
AMM2	16/8	10/1/0.1V**	576-2 – 16-bit, 50kHz A/D (4-1/2 digit)

Expansion slot: Additional channels or signal conditioning.

Module	Channels	F.S. Ranges*	Notes
AIM3A	32/16	10/1/0.1/0.01V**	Full-speed scanning at all gains
AIM4	4	5/0.5/0.05/0.005V	Isolated inputs
AIM5	4	0.05/0.005V***	Isolated inputs, V or thermocouple.
AIM6	4	0.1/0.03V	100Ω RTD (-200 to 700°C)
AIM7	16	0.1/0.01V***	V or thermocouple
AIM8	4	10/1/0.1/0.01/0.001V	V or strain, with excitation
AIM9	2	Sensor-dependent	LVDT, with AC excitation

*Ranges result from combined global and local gain. 576 automatically applies highest appropriate local gain, with reminder as global gain. Available gains vary by module.

**Also measures current using shunt resistor. See 576 cover overlay for connections.

***Thermocouple modules have reference junction; accept TC types J, K, S, T, B, E, and R. Temperature range varies with TC type.

Gain	F.S. Range
1	0-10, ±10
2	0-5, ±5
5	0-2, ±2
10	0-1, ±1
20	0-0.5, ±0.5
50	0-0.2, ±0.2
100	0-0.1, ±0.1

Gain	F.S. Range
200	0-0.05, ±0.05
500	0-0.02, ±0.02
1000	0-0.01, ±0.01
2000	0-0.005, ±0.005
5000	0-0.002, ±0.002
10000	0-0.001, ±0.001
–	–

ANALOG OUTPUT

Standard: Two 50kHz, 13-bit D/A channels, ±10/±5/±2/±1V ranges with 5µs settling time.
Expansion slot: 1 to 5 additional analog output channels with optional module.

Module	Channels	F.S. Ranges	Notes
AOM1	2/5	±10/±5/±2.5/±0-10/0-5V	12-bit D/A
AOM2	1/2	±10/0-10V	16-bit D/A
AOM3	4	0-20mA	12-bit current loop output
AOM4	4	0-10V	12-bit, 40mA current capacity
AOM5	4	±10/±5/±2/±1V	13-bit D/A

DIGITAL OUTPUT

Standard: 32 TTL-compatible channels, configurable in groups of 8 for input or output.
Expansion slot: 48 or 64 total digital channels with optional module.

Module	Channels	Function
DIM1	16	Opto-isolated digital inputs
DIO1	32	4 x 8-channel TTL input/output ports
DOM1	16	Opto-isolated digital outputs

POWER CONTROL

Standard: 16 of the 32 standard digital channels can be used to control or sense external loads from 12-280VAC @ 3A or 10-200VDC @ 3A using the PCM3 accessory.
Expansion slot: 4 additional power control channels with optional module.

Module	Channels	Function
PCM1	4	Control external loads to 140VAC

FREQ/PULSE/EVENT COUNTING

Expansion slot: up to 8 frequency or event counting channels with optional module.

Module	Channels	Function
PIM1	8	Event count/freq count to 8MHz
PIM2	4	4 x 16-bit or 2 x 32-bit event counters

SECTION 2 - System Set-up

Rear Panel Switches

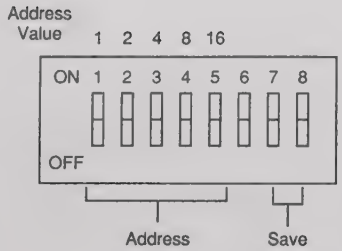
Model 576 contains an 8-position DIP switch for setting system address and the operation of the battery-backed data/program memory.

Switches 1-5 set the primary GPIB address.

Default address = 3: SW 2 ON + SW 1 ON = 2 + 1 = 3

Switches 7 and 8 enable the SAVE command which controls retention of programs and data in memory when the 576 is powered down. See manual for details.

Default SAVE mode = OFF. (Switches 7 and 8 OFF)



Switches 6 is reserved for future use.

Jumper Positions (refer to Model 576 top cover overlay)

AMM jumpers - J3, jumper pins 1 and 2 to enable hardware triggering of analog input by Model 576 trigger circuitry.
576 jumpers - W201, jumper pins 1 and 2 to configure trigger circuit.
AMM-to-576 cable - Attach from AMM J7 to 576 J201 - connects Model 576 trigger output to trigger input of AMM module.

Power Connection

Rear-mount connector for 12-18V DC or AC. External transformer power supply.

Signal Connections

Detachable quick-connect blocks for all I/O connections. See Model 576 top cover overlay for pin-out information.

Quick-Start Instructions and Programs

Instructions and example program disk for fast system set-up are located with the 576. See manual for set-up instructions and listings.

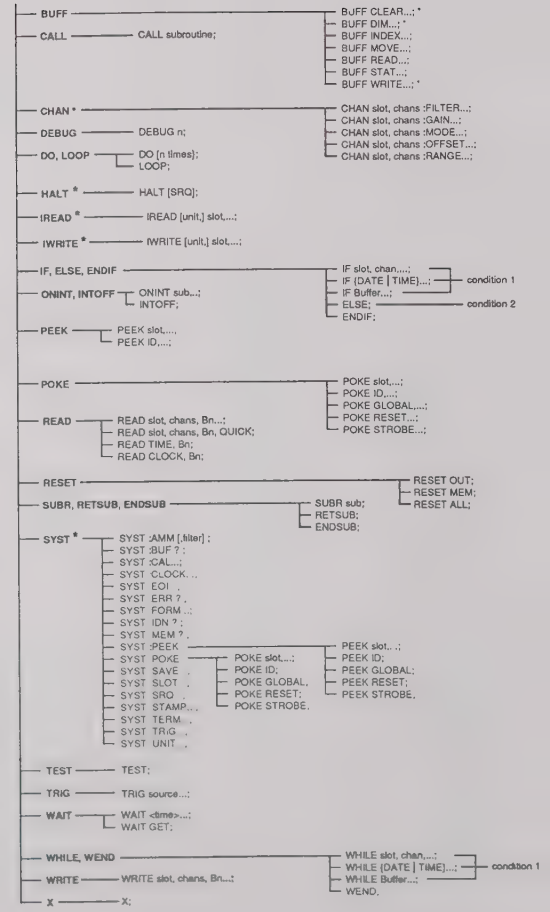
SECTION 3 - Model 576 Commands

Terms, Abbreviations, and Conventions

1. All 576 commands are followed by a space. Parameters are separated with commas.
2. Optional parameters are enclosed in square brackets "[ ]" and are position dependent.
3. The colon ":" combines several functions in a single command without requiring a reissue of the command each time.
4. Braces "{ }" denote that one, but not both, of the enclosed items must be selected. A vertical bar " | " indicates that one or more of several listed options must be chosen.
5. "< >" denotes user supplied values or data.
6. All 576 commands must be terminated with a semi-colon (;).
7. Commands can be delimited with a space, comma, or tab.
8. Only the first 4 characters of any command or function are significant (more than 4 letters may be shown for clarity). If shorter than 4 characters, use the full name.
9. The 576 accepts commands upper or lower case.

10. C = Celsius
    F = Fahrenheit
    EU = Engineering Units Conversion
    SE = Single-Ended (analog input)
    DF = Differential (analog input)
    85 = RTD with alpha of 0.385
    92 = RTD with alpha of 0.392

576 Commands



Notes:

1. "..." indicates additional parameters or options.
2. Only first four characters are significant. Commands less than 4 characters long must be used in their entirety.
3. * indicates IMMEDIATE mode command. All other commands are PROGRAM mode commands.
   Immediate commands execute as soon as they are sent to the 576.
   Program commands do not execute until X; is issued, at which time all pending program commands execute.

BUFF CLEAR

Clear buffer contents

Format: BUFF CLEAR Bn;

Parameters:

Bn - any valid buffer number.

Notes:

1. A BUFF READ Bn; command (no channel) automatically clears a buffer.
2. A BUFF READ [channel] Bn; should be followed with BUFF CLEAR if buffer is to receive new data.

BUFF DIM

Dimension Buffer

Format: BUFF DIM [type,] Bn, #chan, #scan [, STAMP];

Parameters:

type - (optional) array type specifier. Types and usage as follows:
BYTE (8-bit) - read digital port.
WORD (16-bit, default) - read non-thermocouple analog channel.
TC (16-bit, temperature array) - read thermocouple and cold junction (1 word/TC read plus 1 word/per scan for cold junction).
LONG (32-bit) - read pulse input module, time, or clock.

Bn - number of the buffer to be allocated (B0-B19).

#chan - number of channels per scan.

#scan - number of scans per channel to allocate for the buffer.

STAMP - (optional) flag to enable time stamping only into specified buffer.

Notes:

1. For TC measurements use array type "TC" to allocate additional WORD per scan for storing CJR reading (AIM5, AIM7).
2. To READ TIME, buffer must be type "LONG" and one channel. For CLOCK, type is "LONG" with "#chans" of 2.
3. To conserve memory, specify optional array type for PIM or digital I/O arrays.
4. Timestamping automatically allocates 6 additional bytes per buffer or scan (depending on the mode and type) if the optional STAMP flag is specified.

BUFF INDEX

Report point last accessed in buffer

Format: BUFF INDEX Bn;

Parameters:

Bn - any valid buffer number (B0 - B19).

Notes:

1. Buffer must exist or error will be generated.

BUFF MOVE

Move, change format, scale, or offset buffer contents

Format: BUFF MOVE slot, chans, Bnew, Bold [,<scale>, <offset>];

Parameters:

slot - slot number of the module which is the destination for the translated data.
chans - channel or range of channels that specify data conversion (specify range as "start chan - stop chan", e.g. "1-5").
Bnew - number of the buffer to store the translated data (B0-B19).
Bold - number of the buffer to translate (B0-B19).
scale - (optional) scale factor applied to the buffer data during conversion.
offset - (optional) offset factor applied to the buffer data during conversion.

Notes:

1. "Bold" and "Bnew" must be dimensioned the same.
2. Scale and offset are single precision, floating point.
3. If scale or offset are out of range for the device, value is clipped at the legal maximum or minimum for the device.
4. If values are scaled down, rounding errors may result in loss of resolution of original signal.
5. Offset value is in volts only.
6. BUFF MOVE will not accept buffers dimensioned with "STAMP".

BUFF READ

Read contents of entire buffer

Format: BUFF READ [unit,] [chan,] Bn;

Parameters:

unit - (optional) EU specifier, overrides system default if specified.
RAW = raw A/D counts no EU conversion
DCV = volts
MA = milliamperes
TCn [,C | F] = TC type n (J, K, S, T, E, B, or R). Reading in C or F.
RTDn [,C | F] = RTD type n (85 or 92). Reading in C or F.
HZ = Hertz

Bn - number of the buffer (B0-B19).

Notes:

1. Default conversion is RAW. Default temperature is C.
2. Data is read from the buffer according to specified EU, or system default if EU is not specified.
3. If specified buffer does not exist an error will be generated.
4. Format for data read from a buffer that was configured for timestamping:

Mode,Type	Output Format
ONCE,TIME	TIME,<scan1 data><scan2 data>...
ONCE,CLOCK	DATE,TIME,<scan1 data><scan2 data>...
SCAN,TIME	TIME,<scan1 data>,TIME,<scan2 data>...
SCAN,CLOCK	DATE,TIME,<scan1 data>,DATE,TIME,<scan2 data>...

5. Time stamp info returns in ASCII format except for EU of "RAW" (see manual for data formats).
6. See also: SYST :UNIT and SYST :FORMAT.

BUFF STAT

Calculate buffer statistics

Format: BUFF STAT [unit,] [chan,] Bn;

Parameters:

unit - (optional) Engineering Unit. TIME and CLOCK are invalid.
chan - (optional) channel to use. Whole buffer used if channel isn't specified.
Bn - valid buffer number (B0-B19).

BUFF WRITE

Write data to buffer

Format: BUFF WRITE [unit,] slot, chans, Bn, <data>, <data>,...<data>;

Parameters:

unit - (optional) EU specifier; system default if not specified:
RAW = raw A/D counts no conversion
DCV = volts
MA = milliamperes

slot - slot number of the module the data is for.

chans - channel or range of channels (specify range as "start chan-stop chan", e.g. "1-5").

Bn - number of the buffer to be written (B0-B19).

<data> - stream of data sent to the specified buffer.

Notes:

1. Data is converted from the specified data format and EU to raw binary based on the configuration of the specified channels.
2. If specified buffer does not exist an error will be generated.

3. Number of samples is specified with BUFF DIM command. Error results if a line terminator is received before all data is transmitted.
4. See also: SYST :UNIT, SYST :FORMAT.

CALL

Format: CALL subname;

Parameters:
subname - name of a user defined subroutine.

Notes:
1. Subroutine must exist at run time or an error will result.
2. See also: SUBR, RETSUB, ENDSUB, ONINT and INTOFF.

CHAN :FILTER

Set filter for the specified channel

Format: **CHAN slot, chans :FILTER val;**
CHAN slot, chans :FILTER ?;

Parameters:

slot - slot number of the module to associate with the filter setting.
chans - channel or range of channels to associate with the filter setting (specify range as "start chan-stop chan", e.g. "1-5").
val - filter value.
? - returns the current filter setting for the slot/chan as "FILTER [chans] val[, val]...".

Notes:

1. Assigned filter value remains in effect until another filter command is issued or the 576 is reset.
2. AMM modules have global filters. See SYST :AMM for configuring AMM filter.
3. For wideband filter on any module, specify "NONE" for val.

Module	Loc. Filter	Val
AIM 8	1kHz	1K
	10kHz	10K
	Wideband [1]	"NONE" [2]
AIM 9	2Hz	2
	20Hz	20
	200Hz[1]	200
TRG1	300Hz	300
	1kHz	1K
	3kHz	3K
	10kHz	10K
	30kHz	30K
	100kHz	100K
	300kHz	300K
	1MHz[1]	1M
Note [3]	nHz	n
	nkHz	nK
	nMHz	nM

1 Default
2 Use "None" (without quotes) for wideband filter.
3. General format, where filter information for module specifies n Hertz, n kHz, or n MHz. See module manuals for available filters.

CHAN :GAIN

Set total channel gain

Format: **CHAN slot, chans :GAIN total gain;**
CHAN slot, chans :GAIN global, local;
CHAN slot, chans :GAIN ?;

Parameters:

slot - slot number of the module to associate with the gain setting.
chans - channel or range of channels to associate with the gain setting (specify range as "start chan-stop chan", e.g. "1-5").
total gain - value of the total gain to be applied to the specified channels.
global - (opional) value of global gain applied to analog input channels.
local - (optional) local gain applied to analog input channels.
? - returns current gain setting for slot/chan as "GAIN [chans] val, local [val, local]...".

Notes:

1. Default gain switch is X1 on all slots where applicable.
2. Total gain = Local (module) Gain x Global (A/D) Gain. Local gain is automatically set to maximum appropriate value. Remainder of gain is applied as Global.
3. Local and Global gains can be used to override gain values normally set by the system. Global and local gains must by correct or an error will result.
4. For modules with switch selectable gain, "local gain" parameter must be issued.

Module Gains and Input Ranges				
Module	Global	Local	Default	Voltage Range[1]
AMM1A AMM2	1, 2, 5, 10	1, 10	1	10/1/0.1
AIM2	AMM global gain may be applied to any analog input channel	1	1	10/1
AIM3A		1, 10, 100	1	10/1/0.1/0.01
AIM4[2]		1,100	1	5/0.5/0.05/0.005
AIM5[2]		100	100	0.05/0.005/TC
AIM6		50	50	1000Ω RTD Module
AIM7		100	100	0.1/0.01/TC
AIM8		1, 10, 100, 1000	1	10/1/0.1/0.01/0.001
AIM9[3]		1	1	LVDT/RVDT Module
TRG1[4]		1, 10	1	10/1/0.1
All Analog Input Modules	When specifying **Total Gain**:[5] Total = Local x Global		Same as above	Same as above

Notes:

1. Voltage shown at gains of x1, x10, ... x 10,000 as is available with a given module. Other intermediate ranges are available using global gains of x2 or x5.
2. Isolated input. Maximum input = 5V for AIM4, 0.05V for AIM5
3. Module gain continuously adjustable from x1 to x20. Firmware always treats local gain as x1.
4. When TRG1 is used as a single analog input channel.
5. Where several combinations equal the same total, highest appropriate local gain will be used, with any remainder applied as global.

CHAN :MODE

Set operating mode of module (single-ended/differential, gated/normal, read/read&reset, input/output

Format: **CHAN slot, chans :MODE mode;**
CHAN slot, chans :MODE ?;

Parameters:

slot - slot number of the module to associate with the mode setting.
chans - channel or range of channels to associate with the mode setting (specify range as "start chan-stop chan", e.g. "1-5").
mode - operating mode for module in the indicated slot.
? - returns the current mode setting for the slot/chan as "MODE [chans] mode [,option] mode [,option]..."

Module	Mode
AMM1A, AMM2, AIM3A	SE or DF
DIO1, DIO1A	IN or OUT
PIM1	NORM,{CONT | RESET} GATE,{CONT | RESET} FREQ
PIM2	CONT,{16 | 32 } RESET,{16 | 32 }

Notes:

1. Default on power up = system default conditions.
2. PIM2 32-bit mode links channels 0-1 and 2-3. Assigned mode is used by the pair. In 16-bit mode, each channel may have individual mode settings.

CHAN :OFFSET

Enable or disable offset (AIM8 ONLY)

Format: **CHAN slot, chans :OFFSET option;**
CHAN slot, chans :OFFSET ?;

Parameters:

slot - slot number of the module to associate with the offset setting.

chans - channel or range of channels to associate with the offset setting (specify range as "start chan-stop chan", e.g. "1-5").

option - condition of the offset:
ENABLE = enable offset
DISABLE = disable offset

? - returns current offset for the slot/chan as "OFFSET [chans] option option...".

Notes:

1. Default on power up is DISABLE.

CHAN :RANGE

Select range of an A/D, D/A, or PIM1 module

Format: **CHAN slot, chans :RANGE range;**
CHAN slot, chans :RANGE freq;
CHAN slot, chans :RANGE ?;

Parameters:

slot - slot number of the module to associate with the range setting.

chans - channel or range of channels to associate with the range setting (specify range as "start chan-stop chan", e.g. "1-5").

range - operating range for the indicated module:

Analog Input/Output -
10B = ±10V (Default for AMM and AOM5)
10U = 0-10V
5B = ±5V
5U = 0-5V
2B = ±2.5V or ±2V
1B = ±1V

Frequency (PIM1) -
62K = 62 kHz (Default)
125K = 125 kHz
250K = 250 kHz
1M = 1 MHz
2M = 2 MHz
4M = 4 MHz
8M = 8 MHz

? - returns the range as "RANGE [chans] range/freq..."

Notes:

1. On power up all ranges are set according to system defaults

Format: DEBUG number;

Parameters:

number - 0 to 7. DEBUG stores this number as bits 0, 1, and 2 of the serial poll byte when the DEBUG command is encountered in a program. Bits 0, 1, and 2 of the serial poll byte report the last debug number that was set. bit 0 x 1 + bit 1 x 2 + bit 3 x 4 = debug number.

Notes:

1. Numbers assigned to DEBUG need not be sequential or start with 0.
2. DEBUG may slow execution slightly.

DO, LOOP

Format: **DO [n];**
LOOP;

Parameters:

n - loop count, 1 - 4,294,967,295, or loop forever if n is not specified.

Notes:

1. Every DO command must have an associated LOOP or an error will be issued.
2. Loops can be nested to 8 levels.

HALT

Halt program

Format: HALT [SRQ];

Parameters:
HALT - stop program execution. HALT SRQ; stops execution when an SRQ is issued.

Notes:
1. HALT stops program execution.
2. See also: SYST :SRQ

IF, ELSE, ENDIF

Conditional execution

Format: **IF slot, chan, [NOT,] cond, <val1> [[, <val2>] [,unit]];**
IF {DATE|TIME}, [NOT,] cond, <val1>;
IF Bn, [NOT,] FU;
(action 1)
ELSE;
(action 2)
ENDIF;

Parameters:

slot - location of the module being tested.
chan - number of the channel being tested.
NOT - (optional) test for condition being NOT TRUE.
cond - expression which must evaluate TRUE.

Expression	Operator
LT	Less than
GT	Greater than
EQ	Equal to
LE	Less than or equal to
GE	Greater than or equal to
FU	Buffer Full
BT	Between
OR	Test any bits set in val
AND	Test all bits set in val

<val1> - typical value of expression when a single limit or low threshold value is required. If DATE or TIME is used, val1 is the date (in the last format specified), or the time.

<val2> - (optional) typical value of expression used as an upper limit or high threshold value.

unit - (optional) EU conversion. Val1 and val2 will be used in the "units" specified, otherwise the default units will be used.

Bn - number of the buffer to be tested (B0-B19). Test for full (FU) or not full (NOT,FU).

Notes:

1. IF can be nested 8 levels deep.
2. Every IF must have a matching ENDIF or errors will result.
3. Specify date in the same mode used to set the real time clock. For conditional triggering on date+time, use two nested IF statements.
4. With Bit Test Operators the val1 parameter specifies the bits to be tested.
5. See also: SYST :CLOCK, SYST: UNIT.

IREAD

Immediate data read

Format: IREAD [unit,] slot, chans [,<avg>];

Parameters:

unit - (optional) EU specifier. System default used if units not specified.
RAW = raw A/D counts, no conversion.
DCV = DC volts
TCn [C | F] = TC type n :J,K,S,T,B,E or R. Reading in C or F.
RTD n[C | F] = RTD type n :85 or 92. Reading in C or F.

slot - slot of the module being read.

chan - channel or range of channels (specify range as "start chan-stop chan", e.g, "1-5").

Notes:

1. If channel list or engineering units is invalid, an error will result.
2. Typical average command: IREAD 1,0,100 will take 100 readings from channel 0, sum the readings, divide by 100, and store the result in the output queue.

IWRITE

Immediate data write

Format: IWRITE [unit,] slot, chans, <data> ...;

Parameters:

unit - (optional) EU specifier. System default used if units not specified.
RAW = raw D/A counts, no conversion.
DCV = DC volts
MA = milliamperes

slot - slot number of the module to associate with the write.

chan - channel or range of channels (specify range as "start chan-stop chan", e.g, "1-5").

Notes:

1. If invalid channels or units are used, error will result.
2. Data value must be specified for each channel in the list or an error results.

ONINT, INTOFF

Setup background subroutine

Format: **ONINT sub [, <rate>, unit];**
INTOFF;

Parameters:

sub - name of one of ten user-definable subroutines to be executed at the specified ONINT rate.
rate - magnitude of the interrupt period.
unit - time frame of the specified rate:
USEC = microseconds, 125 - 65535
MSEC = milliseconds, 1 - 65535
SEC = seconds, 1 - 3267
MIN = minutes, 1 - 65
MLHZ = millihertz, 1 - 65535
HZ = hertz, 1 - 8000

Notes:

1. The last issued ONINT subroutine is executed at the specified rate. For each new ONINT, the associated "sub" becomes the active background subroutine.
2. "INTOFF;" suspends background activity by turning off interrupts. Subsequent ONINT command resumes execution of the specified routine.
3. Interrupt rate must allow adequate time to execute the specified subroutine or an interrupt overrun error occurs. See manual for background time testing.
4. Any new interrupt rate takes effect on occurrence of the next interrupt. Any new subroutine executes on the next interrupt.
5. READ...,QUICK suspends any interrupts until the READ is complete.

PEEK
Read data at address

POKE
Write data to address

Format: **PEEK slot, cmd, Bn [, FILL];**
PEEK ID, Bn [, FILL];

POKE slot, cmd, {Bn [,cycles]|<val>};
POKE GLOBAL, {Bn [,cycles]|<val>};
POKE RESET, {Bn [,cycles]|<val>};
POKE STROBE, {Bn [,cycles]|<val>};

Parameters:

slot - slot number of the module to be read.
cmd - command to be used for the read (A, B, C, or D).
Bn - number of the buffer to use (B0-B19).
cycles - number of cycles through the buffer. If not specified, one data value is written.
FILL - (optional) flag which specifies PEEK UNTIL BUFFER FULL.
ID - contents at the self ID address location
GLOBAL - contents at the GLOBAL 1 location.

RESET - contents at the RESET location. (GLOBAL 2).
<val> - byte value from 0 - 255.
STROBE - contents at the GLOBAL STROBE location.

SLOT	FUNCTION	COMMAND
1	Option slot	A, B, C, D
2	TRG1	A, B, C
3	Option slot	A, B, C
4	AOM5	A, B
5	DIO1A	A, B

Notes:

1. PEEKing or POKEing invalid locations generates an error.
2. POKEing STROBE, GLOBAL, or RESET strobes the selected line on all modules.
3. Buffers should be dimensioned using "BYTE" type specifier.

READ CHANNELS

Read samples from specified channels and store in buffer

Format: **READ slot, chans, Bn;**
READ slot, chans, Bn ,FILL;
READ slot, chans, Bn, QUICK;
READ slot, chans, Bn, <Avg>;

Parameters:

slot - slot number of the module to be read.
chans - channel or range of channels to be read (specify range as "start chan-stop chan", e.g. "1-5").
Bn - number of the buffer to use (B0-B19).
FILL - (optional) flag specifies READ UNTIL BUFFER FULL
QUICK - flag for high speed READ.
Avg - number of points to be averaged.

Notes:

1. If specified buffer does not exist an error results.
2. If collected data exceeds buffer size, excess data is lost.

3. When type "TC" is specified in BUFF DIM, 576 automatically reads the cold reference junction on the AIM5 or AIM7 once per scan.
4. QUICK option can not be used if "TC" is specified.
5. QUICK option is only valid with AMM1A, AMM2, AIM2, and AIM3A if multiple channels are specified. Errors will be generated for other modules.
6. QUICK option suspends the ONINT command and forces READ to run to completion before resuming program execution.
7. QUICK option can be queued with hardware trigger (TRIG).
8. QUICK option rates: 50KHz on AMM2, 62.5 KHz on AMM1A. With timestamping or hardware triggering: 25KHz/31.25KHz.
9. See also: BUFF DIM, SYST :STAMP, and READ.

READ TIME
Read time only and store in buffer

READ CLOCK
Read date and time and store in buffer

Format: **READ TIME, Bn;**
READ CLOCK, Bn;

Parameters:
Bn - number of the buffer to use (B0-B19).

Notes:
1. CLOCK option requires the buffer to be dimensioned for 2 channels.
2. See also: BUFF DIM, SYST :CLOCK.

RESET

Reset system

Format: RESET mode;

Parameters:

mode - specifies what is to be reset:

OUT = clear analog and digital outputs as follows:
DOM1 outputs to logic 1 (DOM1 has inverting outputs).
DIO1A port A and B to logic 0.
DIO1A port C and D to logic 1.
AOM outputs to 0V.

MEM = clear all data memory to free space list.

ALL = simulated cold boot to power up default state.

Notes:

1. RESET ALL; clears the input buffer and program memory, also resets CHAN and SYSTEM commands to defaults. Any pending commands are lost.
2. If the system is not programmed to reflect the actual switch settings of AOM modules, RESET OUT may produce non-zero output.
3. See also: SYST :RANGE.

SUBR, RETSUB, ENDSUB

Define a subroutine, exit a subroutine before it ends, terminate a subroutine definition

Format: **SUBR sub;**
RETSUB;
ENDSUB;

Parameters:

sub - name of the subroutine to be created (up to 8 characters, longer names are truncated to 8). All characters are converted to upper case. Legal characters are A-Z, 0-9, "_", and "$".

Notes:

1. There can be up to 10 subroutines at one time in the system.
2. Subroutines can call other subroutines but cannot be nested. A subroutine cannot be defined within a subroutine.
3. See also: CALL and ONINT.

SYST :AMM

Set up AMM system filter

Format: **SYST :AMM filter;**
SYST :AMM ?;

Parameters:

filter - AMM filter:
100K = 100 kHz filter
2K = 2 kHz filter

? - return system filter setting as "AMM, filter"

Notes:

1. Default AMM filter is 100K.

SYST :BUF

Return buffer status information

Format: **SYST :BUF ?;**

Parameters: (None)

Notes:

1. SYST :BUF? returns status of system buffers as 4 bytes.
2. Each bit position corresponds to a buffer number. Bit 0 is always the least significant bit. Bit 0 is buffer 0, bit 1 is buffer 1, etc. 1 = buffer FULL, 0 = buffer NOT FULL. Unused bits 20 through 31 are always 0.

SYST :CAL

Setup system AMM module calibration

Format: **SYST :CAL [<const>];**
SYST :CAL ?;

Parameters:

const - factory-supplied calibration constant found on the AMM module. If not specified, the 576 generates the calibration coefficients.

? - returns calibration constants as "CAL..."

Notes:

1. Gain and offset error correction are performed only if SYST :CAL is issued with the calibration constant.
2. To generate correction coefficients for the 576, issue "SYST :CAL;" without the calibration constant after the 576 has met the specified warm-up period.
3. See manual for usage example.

SYST :CLOCK

Setup or read the real time clock

Format: **SYST :CLOCK [format,] {[date,][time]};**
SYST :CLOCK DISABLE;
SYST :CLOCK ?;

Parameters:

format - format of the date string.
STD = MM/DD/YYYY or MM-DD-YYYY (default)
EURO = DD.MM.YYYY
MIL = DD-MMM-YYYY

date - valid parameters for data:
MM - month value between 1 to 12.
DD - day value between 1 and 31.
YYYY - year between 1990 and 2089
MMM - first 3 characters of the month.

time - valid parameters for time. Time is always in 24 hour format - HH:MM:SS
HH - hours 0 to 23
MM - minutes 0 to 59
SS - seconds 0 to 59

? - read the real time clock and return the data as "date time", using the last set format or the default format.

DISABLE - disable the real-time clock oscillator (conserves battery life). Re-enable clock by issuing time/date.

Notes:

1. Default date format is STD, (MM/DD/YYYY)
2. Real time clock will not be set unless the SYST :CLOCK command is issued.
3. See also: BUFF DIM, IF, READ, SYST :STAMP, SYST :TRIG and WHILE.

SYST :EOI

Enable/disable End or Identify flag

Format: **SYST :EOI option;**
SYST :EOI ?;

Parameters:
option - specifies whether EOI is enabled or disabled
ENABLE = enable the transmission of EOI (default).
DISABLE = disable the transmission of EOI
? - returns the status of EOI usage as "EOI ENABLE" or "EOI DISABLE"

Notes:
1. EOI is enabled at power-up.

SYST :ERR

Return system error to controller

Format: SYST :ERR ?;

Command returns error string- short ASCII string which gives the error message.

Notes:

1. Returns system error as "error string".
2. If no errors are active, the string "No system error" is returned.
3. See also: DEBUG.

SYST :FORM

Set system default data transfer format

Format: **SYST :FORM format;**
SYST :FORM ?;

Parameters:

format - data transfer format specifier:
MOTO = Motorola binary format
INTL = Intel binary format
ASCI = ASCII with prefix (Default)
ASCN = ASCII without prefix

? - returns the current default data transfer format as "FORMAT format"

SYST :IDN

Return system identification string

Format: SYST :IDN ?;

Parameters: None

Notes:
1. IDN ? returns the system ID string as "Keithley Inst. Model 576, Rev x.xx".

SYST :MEM

Return size of remaining data and program RAM

Format: SYST :MEM ?;

Parameters: None

Notes:

1. MEM? returns size of the largest block of free data RAM and remaining program memory as MEMORY <bytes of data memory, bytes of program memory>.

SYST :PEEK
Immediate read of 576 address location

SYST :POKE
Immediate write to 576 address location

Format:

SYST :PEEK slot, cmd;	SYST :POKE slot, cmd <val>;
SYST :PEEK ID;	SYST :POKE ID <val>;
SYST :PEEK GLOBAL;	SYST :POKE GLOBAL <val>;
SYST :PEEK RESET;	SYST :POKE RESET <val>;
SYST :PEEK STROBE;	SYST :POKE STROBE <val>;

Parameters:

slot - slot number of the module to be read.
cmd - command to be used for the read. Valid commands are A, B, C, or D.
ID - contents at the self ID address location
GLOBAL - contents at the GLOBAL 1 location.
RESET - contents at the RESET location (GLOBAL 2).
<val> - byte value from 0 - 255.
STROBE - contents at the GLOBAL STROBE location.

Slot	Function	Command
1	Option slot	A, B, C, D
2	TRG1	A, B, C
3	Option slot	A, B, C
4	AOM5	A, B
5	DIO1A	A, B

Notes:

1. PEEKing or POKEing an invalid location will generate an error.
2. PEEKing or POKEing STROBE, GLOBAL, ID, or RESET strobes the selected line of all modules.

SYST :SAVE

Set power-up mode for system SAVE command

Format: **SYST :SAVE option;**
SYST :SAVE ?;

Parameters:

option - level of system configuration at power-up:
OFF = execute self-ID for hardware configuration, clear data buffers, reset all outputs to 0 (DEFAULT).
CNFG = use previous hardware configuration, clear data buffers, reset outputs to 0.
DATA = retain previous hardware configuration and contents of data buffers, reset all outputs to 0.
PROG = retain previous hardware configuration, program, and data, reset data pointers to start of data buffers, reset outputs to 0, restart program (auto-restart mode).

? - return the current system save option as "SAVE option".

Notes:

1. Switches 7 and 8 on the 576 mother board enable SAVE. Default for both is OFF (SAVE disabled). If SAVE is issued with feature disabled, SAVE is ignored.

SYST :SLOT

Configure slots with modules

Format: **SYST :SLOT slot, module;**
SYST :SLOT slot ?

Parameters:

slot - slot location of the module being accessed.

module - module type:

Analog In	Analog Out	Digital	Other
AMM1A	AOM1/2, /5	DIM1	PROTO*
AMM2	AOM2/1, /2	DOM1	EMPTY (Default)**
AIM2	AOM3	DIO1	
AIM3A	AOM4	DIO1A	
AIM4	AOM5	PCM1	
AIM5	PCM2		
AIM6	PIM1		
AIM7	PIM2		
AIM8	* Identified, but supported only with PEEK/POKE.		
AIM9	** Applicable only to slots 1 and 3.		

? - return the module in the specified slot as "SLOT slot, module"

Notes:

1. SYST :SLOT is unnecessary for modules with SELF-ID. Error results if a module is assigned to a slot containing a valid, SELF-ID'd module of another type.
2. "EMPTY" clears a slot entry.

SYST :SRQ

Setup service request mask

Format: **SYST :SRQ cond;**
SYST :SRQ ?;

Parameters:

cond - an ORing of the conditions which will cause an SRQ:
BUFF = buffer full
DATA = data Ready to be read from controller
ERR = error
IDLE = 576 Not Busy, program complete
NONE = no SRQ
- return the current SRQ mask as "SRQ cond [,cond...]"

Notes:

1. Default is NONE (do not use SRQ)
2. When SYST :SRQ NONE; is issued, 576 cannot issue a Service Request.
3. NONE overrides all other mask conditions. Data bits in the status poll byte are always set indicating the Service Request conditions.
4. SYST :SRQ overwrites the previous SRQ settings.
5. DEVICE CLEAR resets the SRQ mask to NONE.

SYST :STAMP

Set up system time/date stamping mode

Format: **SYST :STAMP mode, type;**
SYST :STAMP ?;

Parameters:

mode - specifies how a buffer is to be stamped:
ONCE = stamp buffer on initial write only.
SCAN = stamp buffer on each scan.

type - specifies time stamping data:
TIME = stamp with time only
CLOCK = stamp with date and time

? - returns current time stamp settings as "STAMP mode, type"

Notes:

1. If SYST :STAMP is not issued, default mode = ONCE per buffer, TIME only.
2. Acquisition rates above 100Hz may result in multiple scans with the same time stamp data.
3. Time stamping with hardware triggering active increases time stamp resolution to 1 µs.
4. See also: BUFF DIM, TRIG, and READ...QUICK.

SYST :TERM

Set data transfer line terminator sequence

Format: **SYST :TERM option;**
SYST :TERM ?;

Parameters:

option - the type of terminator sequence to be used:
CRLF = carriage return, line feed (default)
LFCR = line feed, carriage return
CR = carriage return only
LF = line feed only
NONE = no terminator

? - return the system terminator option as "TERM option".

Notes:

1. Default is CRLF (Carriage return line feed).

SYST :TRIG

Select mode for start trigger

Format: **SYST :TRIG mode;**
SYST :TRIG ?;

Parameters:

mode - condition that starts a program:
EXEC = begin on X (Default).
GET = begin on GET (Group Execute Trigger)
{[date,][time]} = begin on date, time, or date and time.

Notes:

1. SYST :TRIG determines when the next received program begins execution. Example: to begin program execution on GET, "SYST :TRIG GET" must be executed before the program is downloaded and executed with X;.
2. Default = Begin Program Execution on X;.
3. Date must be specified in the format currently in effect. See SYST :CLOCK.
4. If both date and time are issued, date must precede the time.
5. The SYST :TRIG command is reset to EXEC upon receiving "RESET ALL".

SYST :UNIT

Set up system default engineering unit conversion

Format: **SYST :UNIT unit;**
SYST :UNIT ?;

Parameters:

unit - sets the system default EU conversion
RAW = raw A/D counts no EU conversion
DCV = volts
MA = milliamperes
TCn [,C|F] = TC type n (J, K, S, T, E, B, or R). Reading in C or F.
RTDn [,C|F] = RTD type n (85 or 92). Reading in C or F.
HZ = Hertz

? - returns the system default EU flag as "UNIT unit [,vals]"

Notes:

1. Default conversion is RAW. Default temperature is C.

TEST

Format: TEST;

Parameters: none

Notes:

1. A version of the self test is executed by the 576 on power up.
2. If power-up tests fail, the SRQ LED flashes at 2Hz or 20Hz.
3. If memory tests pass, system returns message "System ROM and RAM - OK".
4. If TEST fails and SRQ LED does not blink, 576 will not respond to commands.

CAUTION: TEST overwrites any existing data and programs.

TRIG

Hardware trigger

Format: TRIG source, <threshold>, mode, coupling, level;

Parameters:

source - input source for the trigger signal
TRG1 = local TRG1 input.
AMM = output of global AMM amp. Specify channel in READ...QUICK.

<threshold> - data value between -10V and +10V to set trigger level.

mode - trigger card operating mode:
LATCH = trigger in latched mode
EVENT = trigger in event mode

coupling - signal coupling:
DC = DC coupling
AC = AC coupling

level - specifies the trigger region.
ABOVE = trigger above the threshold
BELOW = trigger below the threshold

Notes:

1. LATCH can be used with multiple channel scans.
2. EVENT is for use with single channel scans only.
3. If no filter is specified, TRG1 defaults to 1MHz filter.
4. The only valid trigger modes for multiple channels with the READ..QUICK is LATCH. If multiple channels are to be scanned off of a hardware trigger, connect the trigger signal to the local input on the TRG1.
5. 576 allows time stamping in conjunction with hardware triggering. Configure SYST :STAMP and note the following:

TRIG mode	STAMP mode
LATCH or EVENT	ONCE - Stamp on first trigger event SCAN - Stamp each time trigger is true

6. See also: BUFF DIM, CHAN :FILTER, and SYST :STAMP.

WAIT

Time or "GET" Delay

Format: **WAIT <time>, unit;**
WAIT GET;

Parameters:

time - delay before program execution resumes. Ranges:
1 - 65535 for millisecond range,
1 - 3267 for seconds range,
1 - 65 for minutes range.

unit - time units:
MSEC for milliseconds
SEC for seconds
MIN for minutes

GET - program will wait for Group Execute Trigger (GET).

WHILE, WEND

While condition control

Format: **WHILE slot, chan, [NOT,] cond, <val1> [[, <val2>], unit];**
WHILE {DATE|TIME}, [NOT,] cond, <val1>;
WHILE Bn, [NOT,] FU;
WEND;

Parameters:

slot - slot number of the module being tested.
chan - number of the channel being tested.
NOT - (optional) specifier to test for condition NOT TRUE.
cond - expression which must evaluate TRUE.

Expression	Operator
LT	Less than
GT	Greater than
EQ	Equal to
LE	Less than or equal to
GE	Greater than or equal to
FU	Buffer Full
BT	Between
OR	Test any bits in set val
AND	Test all bits in set val

val1 - value typically used in an expression when a single limit or low threshold value is required. The evaluation of val1's format is specified by units. If not specified the current SYST :UNIT default setting is used.

val2 - (optional) value typically used in an expression as an upper limit or high threshold value. The evaluation of val2's format is specified by "unit". If not specified, the current SYST :UNIT setting is used.

unit - (optional) engineering unit conversion specifier. Val1 and val2 will be used in the specified "unit". If "unit" is not specified, the default unit set in SYST :UNIT is used.

Bn - number of the buffer to be tested (B0-B19) as full (FU) or not full (NF).

Notes:

1. Every WHILE must have a matching WEND or an error will be issued upon receiving X;. WHILE's can be nested 8 levels deep.
2. Time and date cannot both be specified in one WHILE command. For conditional triggering on date+time, use two nested WHILE statements
3. Date must be specified in the same mode used to set the real time clock.
4. With Bit Test Operators the val1 parameter specifies the bits to be tested.
5. See also: SYST :CLOCK, SYST: UNIT.

WRITE

Write data

Format: WRITE slot, chans, Bn [, cycles];

Parameters:

slot - slot number of the module to be written.
chans - channel or range of channels to be written (specify range as "start chan-stop chan", e.g. "1-5").
Bn - number of the buffer to be accessed (B0-B19).
cycles - number of cycles for the specified buffer to perform (1-65,535). If not specified, one scan is written.

Notes:

1. If specified buffer does not exist, an error results.
2. If samples written exceeds buffer size, output recycles back to start of buffer.
3. Maximum number of cycles may be less of the #scans specified if the BUFF DIM command is large. Actual number of cycles = 2^{32}/#scans. If the specified number is too large, the 576 generates an error.
4. See also: BUFF DIM.

X

Execute all commands stored in 576 program memory

Format: X;

Parameters: None

Notes:

1. If the program is set to execute on a Group Execute Trigger (GET), time, or date, then X; must still be issued to arm the program to begin execution at the appropriate event.
2. To stop program execution, use "HALT", "HALT SRQ", or "DEVICE CLEAR".
3. See also: SYST :TRIG.

SECTION 4 - System Programming

Language Description

The Model 576 is programmed through an English-like, intuitive language which resides in firmware within the system. The language controls how the Model 576 performs its various system control, acquisition, and data handling functions. The commands are ready for operation as soon as the Model 576 is switched on.

The Model 576 command set consists of simple strings and modifiers. These commands are compatible with the widest range of programming languages, GPIB and RS-232 interfaces, and driver software. Generally, any language/interface which permits a string to be sent or received over a GPIB or RS-232 I/O port will be compatible with the Model 576.

Using the 576 Language with Typical GPIB Drivers

To accomplish a desired task, specific commands must be sent to the Model 576 in the correct order. Multiple commands

may be combined as one command string. The Model 576 must also be polled at the appropriate time to read back data or status information. The method of reading status information depends on the interface. Some GPIB driver software contains unique, specialized commands for status checking. See the manual for your particular interface for details.

GPIB interfaces and software use different methods for communicating commands to the GPIB bus. Two general methods are used in most interfaces:

1. Some GPIB interface drivers treat the GPIB interface as a DOS-addressable device. The test program must contain initializing statements to "open" the device (the GPIB interface and bus). The GPIB bus can then be accessed the same as writing or reading a printer or serial port. Commands are sent to the 576 by writing the desired command string to the device. Data and status information are retrieved by reading the device and then parsing the returned string.

   The following general write/read formats are used in BASIC to access the Model 576:

   ```
   PRINT#< device no. >, < command string$ >    ' Write
   INPUT#< device no. >, < input string$ >      ' Read
   ```

2. Some GPIB interface drivers use a variation of the CALL statement to communicate with the GPIB bus. The general format of the CALL statement is:

CALL <function> (...< parameter list >...)

where:

"function" is an operator which is a part of the GPIB driver.

"parameter list" is a series of parameters required by the function. One of these parameters is the 576 command string.

The exact format, content, and punctuation of PRINT#, INPUT#, or CALL statements depends on the driver software and programming language being used.

Also note that the Model 576 command set includes "IMMEDIATE" and "PROGRAM" commands. Immediate commands executes as soon as they are sent to the Model 576. Program commands do not execute until an Execute (X;) command is sent to the Model 576, at which time all pending program commands execute.

Typical Programming Sequence

The following series of steps describe a workable model for programming the Model 576. It is also helpful to check the Serial Poll Byte each time you access the 576 to determine the "ready" or "error" status of the system.

1. Send SYSTem commands to configure 576 hardware.
2. Send CHANnel commands to configure the I/O channels.
3. Send BUFFer commands to initialize and allocate data buffers.
4. If buffers will be used for output, generate and write data values to appropriate buffers.
5. Determine which program functions require subroutines. Construct and download the subroutines.
6. Download main program.
7. Execute program.

Checking the Serial Poll Byte

The Serial Poll Byte can be checked as needed to determine the status of the 576. The value of any individual bit can be derived by ANDing the value of the Serial Poll Byte with the value of the desired bit. If the result is 0, then the value of the tested bit is also 0.

Example – A read of the Serial Poll Byte returns 37. The status of the bits can be tested as follows:

If the value of any variable b0 through b7 is other than 0, then the value of the corresponding bit is 1 and the condition is TRUE. In the example, b0, b2 and b5 test TRUE, indicating that an Error has occurred, and that the last DEBUG number was 5 (32 + 4 + 1 = 37).

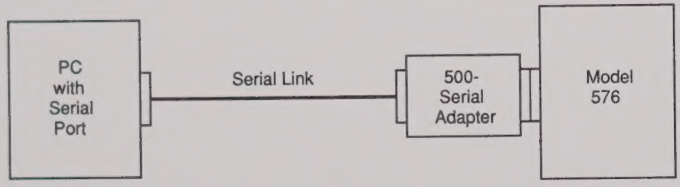
Bit Test Operation on SP%		Bit Function
b0 = SP% AND 1	: If b0<>0 then b0 = 1	'Debug number - bit 0
b1 = SP% AND 2	: If b1<>0 then b1 = 1	'Debug number - bit 1
b2 = SP% AND 4	: If b2<>0 then b2 = 1	'Debug number - bit 2
b3 = SP% AND 8	: If b3<>0 then b3 = 1	'BUFFER FULL bit
b4 = SP% AND 16	: If b4<>0 then b4 = 1	'DATA READY bit
b5 = SP% AND 32	: If b5<>0 then b5 = 1	'ERROR bit
b6 = SP% AND 64	: If b6<>0 then b6 = 1	'SRQ bit
b7 = SP% AND 128	: If b7<>0 then b7 = 1	'IDLE bit

Last encountered DEBUG No. = b0 x 1 + b1 x 2 + b2 x 4

SECTION 5 - Operation with 500-Serial

In addition to operation from a GPIB port, the Model 576 can be controlled through a standard RS-232 serial port by using the Keithley Model 500-Serial converter. The 500-Serial is an RS-232/GPIB converter which plugs into the GPIB port on the rear of the Model 576. A standard serial cable connects the PC's serial port to the 500-Serial adapter.

Instrument capacity: up to 8 GPIB instruments.
Serial connector type: compatible with standard 9-pin or 25-pin serial ports.
Transmission distance: supports GPIB operation at up to 30m from computer.
Transmission speed: baud rates programmable from 300 to 19,200 baud.
Power requirements: derives power from the host serial port. No power supply required.



© 1990 Keithley Instruments, Inc.
Data Acquisition and Control Division
28775 Aurora Road
Cleveland, Ohio 44139
(216) 248-0400

Document Number: 576-903-01 Rev. A